AD-A091 649 ARIZONA UNIV TUCSON INTERACTIVE GRAPHICS ENGINEERING LAB F/G 9/2
SURVEY OF PRE- AND POSTPROCESSING STRUCTURAL ANLAYSIS SOFTWARE.(U)
AUG 80 H A KAMEL N00014-75-C-0837
UNCLASSIFIED TR-7 NL

1 of 1
AD
A091649

END
DATE
FILMED
12-80
DTIC

LEVEL II

ADA091649

N00014-75-C-0837

SURVEY OF PRE- AND POSTPROCESSING STRUCTURAL ANALYSIS SOFTWARE

H. A. Kamel
University of Arizona
Aerospace & Mechanical Engineering Department
Tucson, Arizona 85721

August 5, 1980

Technical Report No. 7

Approved for public release, distribution unlimited

Department of the Navy
Office of Naval Research
Structural Mechanics Program (Code 474)
Arlington, Virginia 22217

Unclassified (14) TR-7

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (9) Technical Report No. 7	2. GOVT ACCESSION NO. AD-A091649	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) SURVEY OF PRE- AND POSTPROCESSING STRUCTURAL ANALYSIS SOFTWARE.		5. TYPE OF REPORT & PERIOD COVERED Technical, 10/24/79
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) H. A. Kamel, H.		8. CONTRACT OR GRANT NUMBER(s) (15) N00014-75-C-0837
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Arizona Interactive Graphics Engineering Lab Tucson, Arizona 85721		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 064-531/12-17-75
11. CONTROLLING OFFICE NAME AND ADDRESS Dept. of the Navy, Office of Naval Research Structural Mechanics Program (code 474) Arlington, Virginia 22217		12. REPORT DATE (11) Augu[redacted]80
		13. NUMBER OF PAGES 21
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same as above (12) 24		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Finite Element Anlysis, Pre- & Postprocessors, Software design.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite Element Analysis, Pre & Postprocessors, Software Design.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The subject of pre- and postprocessing of finite element data has been gaining importance over the last several years. It is perhaps accurate to state that research in this area has been as active as research in the numerical method field. In this context one has to distinguish between several functions associated with what one may call "auxiliary finite element software". Preprocessors perform such functions as geometry definition, geometry

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601

412032 9W

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

discretization (mesh generation), model display and verification, model editing, node and element renumbering as well as load and boundary condition definition. Operations such as load and mass discretization are best considered as part of the analysis. Post-processing may be concerned with result interpretation. Computer graphics and interactive processing play a central role in the activities.

Many commercial organizations are developing and marketing packages aimed at finite element manipulation. Some of these aim at enhancing a specific analysis program. Others are aimed at a specific hardware product, but address popular finite element programs. The third group of programs are both hardware and analysis program independent. There are obvious advantages and disadvantages to each alternative.

Apart from the development of user-oriented pre- and post-processor software, research efforts have been directed at some fundamental points associated with mesh generation and software enigneering. Among the subjects addressed are algorithms for geometry definition, discretization of irregular shapes, and a new and promising subject, grid optimization. The contribution will address the subjects of geometry definition, mesh generation algorithms available as part of postprocessing software, postprocessing and digitizing techniques, mesh optimization, and the design of relevant software. No claim to completeness is being made.

SURVEY OF PRE- AND POSTPROCESSING STRUCTURAL ANALYSIS SOFTWARE

H.A.Kamel, Professor, Aerospace and Mechanical Engineering Dept., University of Arizona, Tucson.

Presented at the Symposium on Mathematical Modeling in Structural Engineering at NASA Langley Research Center on October 24-26, 1979.

INTRODUCTION

The subject of pre- and postprocessing of finite element data has been gaining importance over the last several years. It is perhaps accurate to state that research in this area has been as active as research in the numerical method field. In this context one has to distinguish between several functions associated with what one may call "auxiliary finite element software". Pre-processors perform such functions as geometry definition, geometry discretization (mesh generation), model display and verification, model editing, node and element renumbering as well as load and boundary condition definition. Operations such as load and mass discretization are best considered as part of the analysis. Postprocessing may be concerned with result interpretation. Computer graphics and interactive processing play a central role in the activities.

Many commercial organizations are developing and marketing packages aimed at finite element manipulation. Some of these aim at enhancing a specific analysis program. Others are aimed at a specific hardware product, but address popular finite element programs. The third group of programs are both hardware and analysis program independent. There are obvious advantages and disadvantages to each alternative.

Apart from the development of user-oriented pre- and postprocessor software, research efforts have been directed at some fundamental points associated with mesh generation and software engineering. Among the subjects addressed are algorithms for geometry definition, discretization of irregular shapes, and a new and promising subject, grid optimization. This contribution will address the subjects of geometry definition, mesh generation algorithms available as part of postprocessing software, postprocessing and digitizing techniques,

mesh optimization, and the design of relevant software. No claim to completeness is being made.

In going beyond analysis to the associated automatic model generation and result presentation, the process changes from that of "computer analysis" to "computer-aided analysis". An increase in the computer costs over what is necessary for the analysis itself is to be expected, but the resulting savings in labor, and the faster response, usually more than compensate for the additional expense.

The process of model generation is by no means divorced from that of computer aided design and computer aided manufacturing. It has long been recognized that an integration of design and analysis functions would represent an ideal working environment. Although this is indisputable in principle, its practical implementation is not straightforward by any means, and attempts to construct a formal link between a design data base and analysis programs for complex objects have been typically costly and laborious. It is in the nature of complex systems that an increase in size brings with it a decrease in reliability and flexibility. We may still see some success in such integration attempts, but an acceptable solution will only be possible after significant advances are made in computer software design technology, and more effective use is made of current hardware.

Geometry definition

The process of model definition, in preparation for a finite element analysis, comprises several distinct stages. First a mathematical definition of the geometry is accomplished, and then the discretization process may be carried out. In creating the geometrical model, mathematical methods for the representation of curves and surfaces in three dimensions, typical of those employed in computer aided design may be employed. Some of the methods used in the description of surfaces for analysis purposes are based on mathematics used in finite element computations. A typical example is to be found in isoparametric element formulation. Such methods are usually not as sophisticated as those used in surface design, producing discontinuities in slopes and curvatures at patch boundaries. They are nontheless popular due to the familiarity of the analysts with their formulation, the associated ease of implementation, as well as their ability to describe discontinuous geometry. Although these methods are generally adequate, it has been found in certain applications, such as the analysis of arbitrary shells, that the results may be extremely sensitive to curvatures, thus encouraging the use of the more sophisticated formulations for these purposes.

Curves in three dimensional space may be defined in parametric form[1]. For curves of greater complexity, a piece-wise continuous parametric representation is necessary. Of the many possibilities for parametric representation, the Bezier[2] curves may be mentioned. The Bezier method of surface representation, originally developed for the French automobile industry, derives the parametric shape of a particular curve from a so-called control polygon, connecting n+1 vertices. The resulting parametric curve of order n passes through the two end points, and is tangential to the polygon's first and last sides. The shape of the control polygon influences the shape of the curve. The control exerted by the polygon is global in the sense that a change of the position of one of the points will produce a change of the shape of the curve overall. If relatively few points are used in the polygon, the control of the shape of the curve is good. A Bezier curve is a surface obtained by forming the product of two Bezier curves.

The use of Bezier method presents some problems in the control of the shape of the curves and surfaces, in addition to problems of geometric continuity. Another approach which avoids these shortcomings is that based on spline functions. Such functions involve the fitting of a piecewise continuous curve through a number of points, and provides for continuity of geometric derivatives at segment boundaries through constraint conditions. It has local control, so that the effect of moving a point on the shape of the curve applies only to the immediate neighbourhood of the point. It is possible to form closed or open curves, as in the case of the Bezier method, as well as introduce corners. Cubic splines are the ones most often used.

There exists a number of works which address the question of curve and surface description, and ref.[1] could be consulted for additional detail. For example, Rawat[3], describes the use of bivariate splines in the definition of ship hull geometry. Theilheimer and Mckee[4] survey the use of spline functions in the description of ship hull surfaces. Among the topics covered are the construction of complex curves from control polygons, the computation of surface intersections and the solution of the hidden line problem associated with such surfaces.

Gordon[5] develops a blended spline approach to define complex surfaces using parametric boundary curves. An example of the use of cubic splines as a basis for a mathematical surface definition, followed by a finite element analysis, is given by Liu[6]. He describes an approach by which the smoothness of the surface generated for arbitrary curved shell analysis may be optimized using the unknown twist vectors at the patch corners as controlling parameters.

Mesh generation algorithms.

Many papers exist, which deal with mesh generation algorithms, see for example [7,8,9,10]. an early survey of mesh generation algorithms is found in [31]. In this section we deal primarily with methods for the automatic subdivision of complex geometrical configurations, and the associated computation of the position of intermediate grid points. The simplest and most widely solved problem is the two dimensional one, in which a complex shape, or an interconnected set of arbitrary shapes, is subdivided into triangular or quadrilateral subdomains. Next in degree of difficulty is the problem of definition and discretization of three dimensional surfaces, which may be interconnected. The additional problems arise from the need to determine a method for the definition of the location of the surface in space, as well as the fact that several surfaces may intersect along the same curve. At such intersections surface parameter continuity may be required, although not necessarily. Finally we have the case of three dimensional solid continua, where a volume, bound by such surfaces, may be subdivided into a number of tetrahedron or brick-type elements.

It has been mentioned before that model creation involves both mathematical definition in continuous form, and subsequent discretization into finite elements (subspaces). The procedure by which this may be accomplished is not unique, and may take several forms, as is evident from figure(1). It is possible, for example to generate continuous forms for both curves and surfaces, and then discretize both independently. Since the curves form the surface boundaries, care must be taken that the discretization is compatible. Alternately, one may proceed from a continuous curve directly to a discrete curve format, and use this information to form the discrete surface form without necessarily providing a continuous surface definition. It is also possible to define curves in dicrete form directly, say by digitizing point by point, and then proceed to generate discrete forms for both surfaces and solids.

Another approach evolves from disregarding the natural geometrical hierarchy, and proceeding directly to the definition of the surfaces in continuous form. Once the surfaces have been defined, their intersection is computed, thus defining their boundary curves, as well as the form of the solids bounded by them. All the above approaches have their advantages and disadvantages. A truely general purpose model generation program should allow all possible paths, so that the user may choose whichever approach may best suit his particular problem.

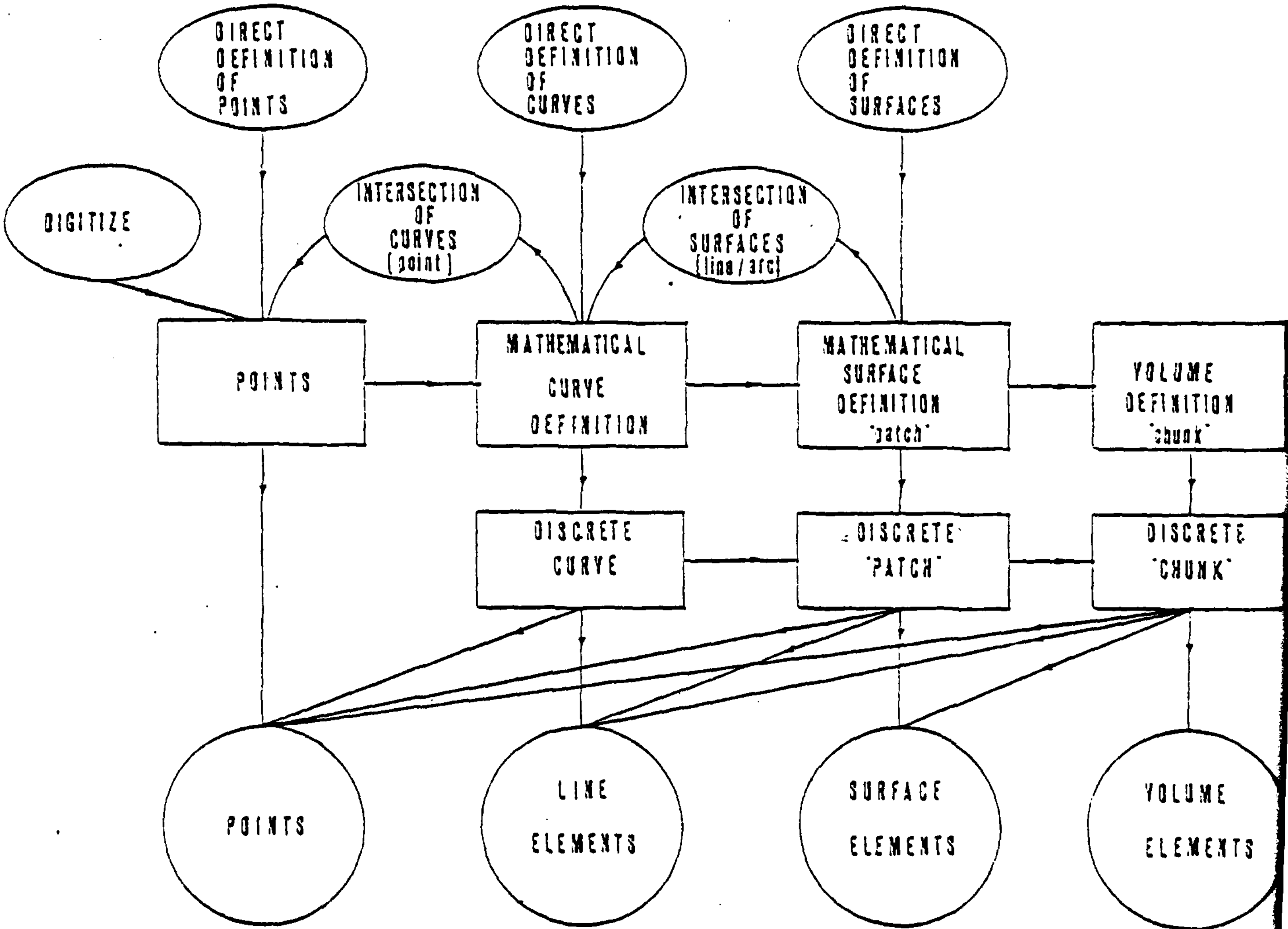

Figure 1. Possible Paths in Model Generation

Turning to mesh generation algorithms, a method for subdividing a two dimensional area, bound by an arbitrary polygon, into triangular finite elements is given by Bykat[11]. The method subdivides the area first into a number of convex subregions. This is followed by an automatic "pre-conditioning" of the boundaries, by introducing additional boundary points to eleminate sudden changes in grid spacing. Automatic boundary refinement is applied at re-entrant corners. Finally all convex regions are triangulated, producing well shaped elements. A two dimensional mesh generator with many interesting features is described by Gabrielson[15]. It is based on the definition of multi-connected four sided regions (i,j grids), in which intermediate points are computed by linear interpolation. It accepts input alphanumerically via the terminal, in the form of data cards, digitizer output or directly from a larger geometric data base intended for computer aided manufacturing. The generated meshes can be used for either heat transfer or stress analysis.

Gordon and Hall[12] address the problem of domain mappings in the context of finite element analysis and present a heuristic method. Nielson and Wixom[13] discuss different techniques for surface definition from boundary curves. Extensions to the method of Gordon are presented. A mesh generator based on B-spline curve representation[14] uses a command language for the generation of points, lines, circles, and various other curve and surface types. It has the ability to calculate surface intersections.

An early example of mesh generators for solid analysis is described in [19]. There, the finite element isoparametric formulation is used as a basis for surface representation and discretization as well as volume discretization. Cook [20,21] develops and uses the concept of "natural coordinates" to generate surface and volume grids. Another effort based on the same concept is presented by Aral[22]. A process for checking solid models by means of computing and displaying the intersections of the elements of the model with a series of planes is described in [23].

Some authors present attempts at developing a language to be used as a means for input description. For example, the SAIL language[24], developed by Boeing is a FORTRAN-like structural input language, which generates input data for many finite element programs, particularly NASTRAN and SAMES. Silvester[25] describes the MAG_NET language, which is a tool for the description of finite element models. It allows geometrical and topological changes, unions of separate models as well as addition and removal of model constituents. Input may be through a keyboard or graphics cursor. It is designed particularly for repetitive structures.

In most programs a bandwidth or wavefront optimizer is necessary to reduce subsequent solution times, and sometimes to avoid exceeding program limitations. A method for renumbering elements in order to achieve improved storage in preparation for a frontal solution, and other methods not requiring a stiffness assembly, is given by Bykat[26]. It represents a modification of the Cuthill-McKee algorithm[27], and requires a reasonable choice of the first element.

Gray et al.[28] describe two methods for automatic renumbering of nodes for water network analysis. The method can be extended to finite element programs. It is aimed at the minimization of matrix fill during the solution process.

Other special efforts deserve attention. An example is an algorithm by Preiss[29], which checks topological consistency of plate and solid models. It is part of an in-house NONSAP preprocessor. In problems where model geometry changes due to chemical reactions or physical phase changes, such as models with abelating boundaries, the problem of automatic model generation is further complicated. Weeks and Cost[30] propose an algorithm for tracking such boundaries, in conjunction with an automatic mesh generator, and a finite element program for the calculation of transient thermal or stress analysis of such sructures.

Model generation programs may be written as general purpose packages or as special purpose programs. A general purpose mesh generator is more versatile, being able to handle a wide variety of problems, and respond well to new and unforseen situations. On the other hand, it will, by necessity, be more complex and will require a certain amount of training before it can be used effectively. A truely general purpose program should impose no restrictions on problem size, or type. If required to run interactively, it will typically be restricted to a small amount of core. This combination of requirements may result in extensive disk usage and computer overhead to perform the necessary I/O operations. The other alternative is that of a special purpose mesh generator, subject to certain restrictions regarding problem type and perhaps size. This option is particularly attractive in situations where a certain type of structure is repeatedly analyzed with different geometric parameters. In such a case it is often possible to produce an optimized grid based on the overall characteristics of the structure in mind, and thereby relieve the user of detailed decisions and input detail. It is often possible, and indeed most desireable, to use an approach combining the two possibilities to best advantage. In this case a truely general purpose mesh generator is coupled with several special "pre-preprocessors", whose functions would be to provide appropriate instructions for the generation of specific structural models of known configuration using the general purpose generator and a minimum of input.

An excellent example of a special purpose mesh generation program is that developed by Leick and Potvin[16]. A special purpose analysis program was written to handle tubular joints such as T,Y,K and TK joints, prevalent in offshore tubular structures. The associated mesh generator produces a fine grid near stress concentration areas and a course one away from them. The grid characteristics are defined in terms of a few parameters.

Available Mesh Generation Software

In general mesh generators may be developed for specific hardware, or may be intended to be hardware independent. Hardware dependent packages are usually associated with turnkey systems, including for example refresh type graphics. In the same manner, pre- and postprocessors may be tailored to a specific analysis program, or may be intended for use by a number of different programs. There are advantages and disadvantages to system dependent packages and system independent ones. Whether the package is dependent on hardware or analysis program or both, it is usually well maintained by the originators. Should the hardware or analysis program change, the auxiliary software is automatically changed and tested. Furthermore, the pre- and postprocessor programs would cater well to the ideosynchronacies of the analysis program, and would utilize special hardware features most effectively. On the other hand, the user's dependence on a single source hampers his freedom of action. Should a particular type of hardware be dropped by the manufacturer, or proves to be ineffective, the required change over to another system is usually costly in both lost production time and manpower. Furthermore, if the auxiliary software is analysis program dependent, and the user has several in-house programs, the personnel training problems may become exessive. In the case of general purpose auxiliary software, the training may be more demanding, but need only be done once for all programs. A maximum flexibility of action is guaranteed. On the other hand, it is more time consuming to maintain a hardware and analysis program independent program. Any change in hardware or analysis program characteristics has to be carefully studied as to its effect on the pre- and postprocessor software. The group performing the maintainance has to have continuous access to all hardware and software in question. This is only possible for extremely large organizations. It may also be accomplished through a well organized users group. A survey of available pre- and postprocessor software is given by Napolitano et al.[32]

Turning to specific software reported in the literature, it is perhaps appropriate to start by mentioning those packages supporting popular analysis software. We start by surveying auxiliary software specifically aimed at particular analysis program. Following that, we shall consider analysis program independent software.

Of all commercial programs, perhaps the one most used in practice is NASTRAN. An early survey of NASTRAN pre- and postprocessors is given in ref[17].The paper lists a number of model generation and bandwidth node resequencing programs available through various organizations.

As mentioned earlier, it is important to preserve the mesh generation geometry and logic for later use in the input of loads and boundary conditions. A pioneering effort to that effect was conducted by Cook[13], who used spline representation of curves and surfaces for geometry definition. The resulting preprocessor was intended for the

NASTRAN program. A general purpose postprocessor for NASTRAN, which operates in both the batch and time-sharing environments is described in ref.[33]. A modelling system designed for NASTRAN is described in [34]. It is based on a hierarchial representation of gemoetry, here called "construction-in-context", and has parametric representation of curves,surfaces and volumes. IGFES[35] is an interactive NASTRAN pre- and postprocessing system, operational on IBM-360 and PDP-11 computers. It is modular and uses storage tube graphics as well as mechanical plotters. It includes both two and three dimensional shell mesh generators and model editors. Amongst mesh generation options are Laplacian, linear interpolation between opposite sides (here called direct ray), Coons blending and isoparametric mapping. It has contouring capabilities.

Several auxiliary software packages have been designed with the popular SAP program in mind. One of the preprocessors is due to Kaldjian[36]. It is interactive, efficient but has somewhat limited capabilities. FEPSAP[37] is an optimized version of the SAP program for CDC computers. It has specialized two dimensional and solids preprocessors. These preprocessors may also be used to generate loads and boundary conditions. Germanischer Lloyd[38] created a system of programs built around the SAP-IV program. Among the available programs is an automatic mesh generator called GLGEN.

Two other programs, particularly popular in Europe are SESAM-69[39] and ASKA[40]. Haugerud[41] describes a solids preprocessor written for SESAM-69. It is based on the creation of a topological arrangement of three dimensional grids, which can then be distorted to fit the physical geometry of the object being analyzed. Automatic node numbering, definition of physical properties and specifications of loads are possible. A further development of the SESAM-69 program is a modular system[42], which includes a finite element pre- and postprocessor utility package, DASA[43]. DASA has approximately 3000 statements and 200 subroutines. It required 4 to 5 man years to develop. It allows the user to implement special and general purpose data generators, using its subroutines as building blocks. It is based on the concept of the "mesh", which is defined by a system of orthogonal surfaces, which may assume prismatic, cylinderical or spherical shapes. The generated model may be modified later to introduce local detail. The system also allows for automatic mesh refinement, as well as load and boundary condition generation.

Uerkvitz[44] and Grieger[45,46] describe a geometrically hierarchial mesh generation program, called INGA, aimed at the finite element program ASKA. It uses finite element interpolation functions as a basis for its mesh generator, and is interactive in nature, using primarily a refresh graphics stand-alone system.

Moving on to program independent auxiliary software, an early example is given by the work of Bousquet and Yates[78]. The previewing and model editing software was intended as an analysis program independent package, and operated on storage tube terminals, thus pointing the way to many subsequent developments. A number of commercially available packages exist, as well as others in the public

domain. The UNISTRUC[47] package uses the so called "drag method" in order to develop three dimensional node and element arrangements from two dimensional ones, and two dimensional ones from one dimensional strings. The FASTDRAW program[48] has been developed by McDonnell-Douglas and serves as a preprocessor for STRUDL, NASTRAN and ANSYS. Tektronix has developed a software package for its line of graphics hardware devices, called FEM. It is designed for a stand alone intelligent device such as the Tektronix FEM-181 system. It has extensive interactive editting capabilities, and a new mesh generation module has been announced.

Several packages of Pre- and postprocessing software have been developed in a University environment as a byproduct of research efforts, or out of necessity to facilitate access to available finite element software. The GIFTS system[50] is one example. It has two and three dimensional model, load and boundary condition generation capabilities. It is designed to run in minimum core space. It's postprocessor handles all but solid models at present. It is supported by a users group, and is distributed out of the University of Arizona. Interfaces exist with the SAP-IV and ANSYS programs, and others are under development. Apart from pre- and postprocessing modules, GIFTS is also capable of static and dynamic analysis on both a number of mainframes and minicomputers. A finite element preprocessor, FEMGEN[51,56], has been under development at the University of Lund in Sweden since 1974. It is intended to support analysis programs such as ASKA, NASTRAN, STRUDL, SAP-IV and ADINA. FEMGEN uses storage tube terminals, drum plotters and line printers. Straight lines and circular arcs may be defined in terms of geometric points, or as the intersections of standard mathematical surfaces, such as planes, cylinders and cones. The program runs in 25K words on a UNIVAC-1100, and has been converted to CDC, IBM and PDP-11 computers.

Many other efforts are underway. For example, the SUPERNET program[52] is being developed in the German Federal Republic by Braun Boveri (BBC). The program supports line, surface and solid elements, and follows a geometric hierarchy. Interfaces are being written for analysis programs such as ASKA, NASTRAN, MARC, TOPAS and ANSYS. SUPERNET may be run in batch, or in an interactive mode. In the latter case a Computer Vision CAD system is used. A pre- and postprocessor package called FEMALE[53] is intended as a general purpose package. However it seems that its mesh generation capabilities are still at an early stage. From the current description it appears to be more of a model pre-viewer rather than a preprocessor. An interactive three dimensional shell mesh generator is described in ref[54]. It uses refresh scope graphics, light pen and keyboard. Initial input, defining the outlines of surface patches is introduced via cards. The user may view the outlines and perform certain operations. Afterwards, the mesh generator may be invoked, and results displayed and editted. A bandwidth optimizer is included, and output may be in the form of punched cards, permanent files or magnetic tape output. Crawford[55]

is in the process of developing a pre- and postprocessor package, which uses a geometrical hierarchy.

Some programs have an integrated pre- and postprocessing capability, which may often be important in special purpose codes, in which the processes of model generation and of computation are interweaved. An interresting example is the TOTAL program reported by Beaubien[57]. It is a two dimensional analysis program designed to simulate the failure mechanisms in layered orthotropic composites. It has a built in pre- and postprocessor which helps keep track of the progress of cracks.

Qualitative evaluation of interactive versus batch computations are hard to find in the literature. Measurements are bound to be system dependent, and often represent personal opinions and experiences. It is nevertheless interesting to mention figures extracted from a promotional publication[34]. It is claimed there that an interactive solution costs only 45% of the cost of a similar computation performed in the batch mode. Out of 100 dollars spent in a batch computation, 70 are spent on model generation, 10 on computer processing and 20 on result interpretation. In the interactive solution, 45 dollars are spent on the computation, 20 of which are spent on model generation, 15 on computer processing and 10 on result interpretation. It is not clear whether these figures are typical of the program in question or are generally valid.

Postprocessing and display techniques.

Although postprocessing is often directly related to the mesh generation process, special display techniques are often employed at this stage, so that it is appropriate to report on postprocessor systems and such techniques under the same heading.

Contouring of stress and deflection results is a useful and popular postprocessing operation. Ref.[58], for example, gives a FORTRAN program which admits several subregions, each divided into triangular or quadrilateral finite elements. The problem of contouring within higher order element boundaries is mathematically intriguing. An initial effort has been conducted by Akin and Gray[59], in which they lay out the fundamental relationships and provide some numerical examples. A follow up paper[60] utilizes a predictor-corrector method to trace such contours accurately. Similarly, Meek and Beer[61] describe a method for tracing deflections across higher order elements. It is based on a process involving averaging nodal values, followed by element subdivision and local linearization. Results are given for triangular and quadrilateral elements.

Akin and Stoddart[62] present an algorithm for plotting intersections of isoparametric solid elements with an arbitrary plane, and the subsequent plots of stress or displacement contours on the resulting mesh. The plots are generated point by point in a random order, and are suitable for graphics terminals or electrostatic plotters. Frey et al. [63] describe a method for calculating the intersection of a plane with second order isoparametric solid brick elements.

A paper describing an algorithm for clipping and capping solid polyhedron[64] is directly applicable to the display of three dimensional models. It produces an input file for the well distributed hidden line program, MOVIE-BYU[65]. A version of the program for 16 bit minicomputers, called MOVIE-ARIZONA, is also available[35]. A two and three dimensional half-tone and color program for the output of finite element results is described by Murai and Tataishi[66]. It has some mesh generation capabilities, is capable of solving the hidden line and surface problems and contouring. Furthermore it uses inexpensive hardware.

Mesh Optimization

Among the most interesting areas of current research in preprocessing is that of mesh optimization. Here the stress is on devising the optimum mesh configuration for a particular element type and an approximate number of degrees of freedom in order to obtain an optimum solution for the problem at hand. It is obvious that a clear definition of what constitutes an optimum solution is necessary. It will be found, for example that high accuracy in a peak stress value, achieved by an optimized grid may be associated with poor deflection results. It is also obvious that a grid may be optimum for one specific loading case, but far from it for another.

Several authors have been active in this area, as is evident from ref.[67,68,69,70,71]. A paper by Turke and McNeice[72] introduces the node coordinates of the mesh as independent unknowns in the finite element variational formulation. The solution, which is based on a minimization of the potential energy of the system, provides both displacement values as well as new node coordinates. Certain constraints have to be imposed on the node relocation to ensure meaningful results. Carroll[73] extends grid optimization to the area of vibrational mode computation. He shows that a different optimum grid is associated with each mode.

Turke and McNeice[74] and Melosh and Killian[75] show the superiority of an optimized mesh over one obtained by uniform refinement. The introduction of additional freedoms and elements was allowed in [75], based on local solutions to study the effect of element subdivision. Denayer[76] generalized some algorithms given in [8] for the generation of topologically uniform grids. The mapping process from a parent grid to the area under consideration is chosen so as to achieve certain optimal conditions, which translate into a variational formulation lending itself to a finite element solution. Carey[77] presents a method for selective refinement of a finite element mesh near an area of interest. Lagrangian constraints are used to maintain continuity with the rest of the model. Results show good convergence in the solution of the Poisson equation with a singularity in two dimensions.

Interactive Software Design

Associated with interactive software, special problems in design methodology, software reliability, portability and testing arise. Some of the surveyed work is being directed to this specific area. Requirements for developing general purpose pre- and postprocessing systems are formulated by Tischler and Bernier[79], and discussed by Herness and Tocher. In [80] it is claimed that 70% to 90% of the cost of a finite element analysis is typically attributed to manpower costs, the rest being computer charges. The concept of a modular program library, in which the pre- and postprocessors interact with the analysis programs via a standard data base is well represented by the GIFTS package[81]. The design of interactive pre- and postprocessor systems is described in [82], which discusses procedures for program design, testing, overlaying, data handling and command language design. The problem of program portability is more complex for graphics software than it is for standard batch analysis programs. Foley[33] discusses the design of device independent basic software.

Conclusion

The paper reviews the principal components of finite element pre- and postprocessing (so-called "auxiliary finite element software"). It provides an assesment of the current state of the art and a literature survey.

Acknowledgements

The author would like to acknowledge the support of the Office of Naval Research under contract number N00014-75-C-0837, and that of the National Science Foundation under contract number ENG77-17313. Miss Maria Pinedo helped in typing the manuscript.

References

1. W.M. Newman and R.F. Sproull, Principles of Interactive Graphics, Second Edition, McGraw Hill, 1979.

2. P. Bezier, Mathematical and Practical Possibilities of UNISURF Computer Aided Geometric Design, R.E. Barnhill and R.F. Riesenfeld (Eds.), Academic Press, 1974.

3. P. Rawat, Use of Bivariate Spline Functions in Preprocessors for Ship Structure Design, Comp. & Struc., Vol. 6, p. 369-374, 1976.

4. F. Theilheimer and J.M. McKee, The Role of Splines in Computer Aided Ship Design. Presented at First International Symposium on Computer Aided Hull Surface Definition, SNAME, Annapolis, Md., Sept. 1977.

5. W.J. Gordon, Blending Function Methods for Bivariate and Multivariate Interpolation and Approximation, SIAM J. Num. Anal., 8 (1971), p. 158-177.

6. D. Liu, Finite Element Analysis of Arbitrary Shells. Ph.D. Dissertation, Univ. of Arizona, Dec. 1978.

7. C.O. Frederick, Y.C. Wong and F.W. Edge, Two-Dimensional Automatic Mesh Generation for Structural Analysis. Int. J. Num. Meth. Engng., Vol. 2, p. 133-144, 1970.

8. H.A. Kamel and H.K. Eienstein, Automatic Mesh Generation in two and three dimensional Interconnected Domains. Symp. on High Speed Computing of Elastic Structures, Liege, Belgium, 1970.

9. O.C. Zienkiewicz and D.V. Phillips, An Automatic Mesh Generation Scheme for Plane and Curved Surfaces by Isoparametric Coordinates. Int. J. Num. Meth. Engng., Vol. 3, p. 519-528, 1971.

10. J.C. Cavendish, Automatic Triangulation of Arbitrary Planer Domains for the Finite Element Method. Int. J. Num. Meth. Engng., Vol. 8, p. 676-696, 1974.

11. A. Bykat, Automatic Generation of Triangular Grid: I-Subdivision of a General Polygon into Convex Subregions. II-Triangulation of Convex Polygons. Int. J. for Num. Meth. in Engng., Vol. 10, p. 1329-1342, 1976.

12. W.J. Gordon and C.A. Hall, Construction of Curvilinear Coordinate Systems and Applications to Mesh Generations, Int. J. Num. Meth. Engng., Vol. 7, p. 461-477, 1973.

13. G.M.Nielson and J.A. Wixom, Approximation Theory Techniques for Curve and Surface Description, 1st International Symposium for Computer-Aided Hull-Definition, p. 107-130, Annapolis Md., Sept. 26-27, 1977.

14. M.E. Golden, Geometric Structural Modelling: A Promising Basis for Finite Element Analysis, Comp. and Structures, Vol. 10, p. 347-350, 1979.

15. V.K. Gabrielson, Mesh Generation for Two-dimensional Regions Using a DVST Graphics Terminal, Sandia Laboratories, Rept. No. SAND76-8231, July 1976.

16. R.D. Leick and A.B. Potvin, Automated Mesh Generation for Tubular Joint Stress Analysis, Comp. and Structures, Vol. 7, p. 73-91, 1977.

17. G.C. Everstine and J.M. McKee, A Survey of Pre- and Postprocessors for the NASTRAN Structural Mechanics Computer Program, W. Pilkey, K. Saczalski, H. Schaeffer (Eds.), U.P. of Virginia, 1974.

18. W.L. Cook, Automated Input Data Preparation for NASTRAN GSC-11039, Goddard Space Flight Center, Greenbelt, Md., April 1969.

19. O.C. Zienkiewicz and D.V. Phillips, An Automatic Mesh Generation Scheme for Plane and Curved Surfaces by `Isoparametric

Coordinates'. Int. J. for Num. Meth., Engng., Vol. 3, p. 519-528, 1971.

20. W.A. Cook, INGEN: A General Purpose Mesh Generator for Finite Element Codes Applications Using ADINA, K-J. Bathe (Ed.), Aug. 77.

21. W.A. Cook, Body Oriented (Natural) Coordinates for Generating Three-Dimensional Meshes. Int. J. Num. Methods, Engng. Vol. 8 27-43 (1974).

22. K. Aral, Data Generation for a 3-D Finite Element System, ASME PVP Conference, Miami, FA, 1974.

23. A.E. Frey, C.A. Hall and T.A. Porsching, An Application of Computer Graphics to Three Dimensional Finite Element Analysis. Comp. and Structure, Vol. 10, p. 149-154, 1979.

24. The ASTRA System, Boeing Document D2-125179-8, May 1972.

25. P.P. Silvester, The MAG-NET-78, Finite Element Model Description Language. U.S. - Japan Seminar on Int. Finite Element Analysis, Cornell University, Aug. 78.

26. A. Bykat, A Note on an Element Ordering Scheme. Short Communication, Int. J. for Num. Meth., Engng, Vol. 11, No. 1, p. 194-198, 1977.

27. E. Cuthill and T. McKee, Reducing the Bandwidth of Sparse Symmetric Matrices, Proc. 24th Nat'l. Conf., ACM, Publication, p. 69, 1969.

28. R.K.L. Gay, K.K. Chin, S.H. Chua, C.H. Chan and S.Y. Ho, Node Reordering Algorithms for Water Network Analysis, Int. J. for Num. Meth. in Engng., Vol. 12, p. 1241-1259, 1978.

29. Preiss, K., Checking of the Topological Consistency of a Finite Element Mesh, Internal report, SRI International, Poulter Laboratory, May 1978.

30. G.E. Weeks and T.L. Cost, An Algorithm for Automatically Tracking Ablating Boundaries, Int. J. for Num. Meth. in Engng., Vol. 14, p. 441-449, 1979.

31. W.R. Buell and B.A. Bush, Mesh Generation--A Survey Journal of Engineering for Industry, Transactions of the ASME, Feb. 1973.

32. G. Napolitano, R. Monti and P. Murino, Preprocessors for General Purpose Finite Element Programs. Str. Mech. Comp. Prgms., W. Pilkey, K. Saczalski, H. Schaeffer (Eds.) U.P. of Virginia, 1974.

33. A.J. Raibstone and A. Pipano, RINA--An Interactive System for the Rapid Interpretation of NASTRAN results, Sixth NASTRAN Users' Colloquium, NASA Conf. Pub. 2018, Oct. 77.

34. E.L. Stanten, An Automated Data Generator for NASTRAN, Sixth NASTRAN Users' Colloquium, NASA Conf. Pub. 2018, Oct. 77.

35. W.E. Lorensan, Interactive Graphics Support for NASTRAN, Sixth NASTRAN Users' Colloquium, NASA Conf. Pub. 2018, Oct. 77.

36. M.J. Kaldjian, Interactive and Data Mode Preprocessor fo SAP, Str. Mech. Series, Vol. 1., N. Perrone, W. Pilkey, B. Pilkey (Eds.) U.P. of Virginia, 1977.

37. D.B. Van Fossen, FEPSAP--Design Program for Static and Dynamic Structural Analysis, Comp. and Structures, Vol. 9, p 371-376, Oct. 1978.

38. C. Nath, Erweiterungen zu SAP IV fuer die Berechnung schiffbaulicher und meeres technischer Probleme, Paper given at the SAP Users' Conf., Bochum, June 1978.

39. O. Egeland and P.O. Araldsen, SESAM-69, A General Purpose Finite Element Method Program, Comp. and Structures, Vol. 4, p. 41-68, 1974.

40. ASKA--Automatic System for Kinematic Analysis, Users' Manual, ISD-Rep. No. 156, 1974.

41. M.H. Haugerud, Interactive 3D Mesh Generation by an Idealization and Mapping Technique. Third International Conf. and Exhibition on Computers in Engineering and Building Design, Brighton, England, March 1978.

42. C. Mo, H.F. Klem, E. Pahle and T. Harwiss, Finite Element Programs Based on General Programming Systems, Comp. and Structures, Vol. 8, p. 703-715, 1978.

43. E. Pahle and K. Flatlandsmo, DASA, Data for Structural Analysis, General Description, Aker Group and Computers, Oslo, Norway, 1976.

44. M. Uerkvitz, Interactive Generation of three-dimensional Idealizations for Finite Element Analysis. Dipl.-Ing, Thesis, U of Stuttgart, 1977.

45. I. Grieger, Interactive Graphical Pre- and Postprocessing for Finite Element Analysis with INGA, Paper presented at the 6th Univac User Association/Europe (UUA/E) Structural Analysis Special Interest Group Meeting. Madrid, Spain, Oct., 1977.

46. I. Grieger, Geometry Elements in Computer-Aided Design, Comp. and Structures, Vol. 8, p. 371-381, 1978.

47. S. Park an C.J. Washam, Drag Method as a Finite Element Mesh Generation Scheme. Comp. and Structures, Vol. 10, p. 343-346, 1979.

48. FASTDRAW Interactive Graphics System, McDonnel-Douglas Automation Co., Document B1298383,1973.

49. J.Z. Gingerich, M.M. Abe, R.L. Vinecore, G.J. Romans, and B.M. Ratihn, A Stand-Alone Interactive Graphics Finite Element Modeling System. Sixth NASTRAN Users' Colloquium, NASA Conf. Pub. 2018, Oct. 77.

50. H.A. Kamel and M.W. McCabe, GIFTS: Graphics Oriented Interactive Finite Element Time-Sharing System. Structural Mechanics Software Series, Vol. I, N. Perrone, W. Pilkey (Eds.) U.P. of Virginia, 1977.

51. T. Johansson, Interactive Preprocessor for Low Cost Hardware Finite Element News, p. 9-11, Oct. 1978.

52. K.E. Buck, U.V. Bodisco, and K. Winkler, SUPERNET, Data Generation for Finite Elemens BBC report, Dept. ZKN/C, Mannheim, June 1977.

53. G.J.V. Shoppee, P.J. Jeanes and T.B. Griffin. A Finite Element Modeling and Analysis Language for Engineering the Program FEMALE, Advances--Engineering Software, Vol. 1 No. 1, p 37-41, 1978.

54. T.J. Dwyer, T. Hasegawa and K.R. Lynn, General-Purpose Structural Graphics System (GPSGS), ASME, PVP Conf., Miami, FA, June 1974.

55. J.E. Crawford, Pre- and Post Processors for General Purpose 3-Dimensional Finite Element Program Application Using ADINA, K-J. Bathe (Ed.), April 1977.

56. T. Johansson, FEMGEN--A General Finite Element Mesh Generator. Application using ADINA, K-J. Bathe (Ed.), April 1977.

57. L.A. Beaubien, TOTAL: Interactive Graphics System for the Two-Dimensional Analysis of Linear Elastic Solids, Str. Mech. Series, Vol. 3, N. Perrone, W. Pilkey, B. Pilkey (Eds.), U.P of Virginia, 1977.

58. E.C. Kalkani, Computer Plotting of Stress Contours in Excavated Slopes. Int. J. for Num. Meth. in Engng., Vol. 10, p. 1261-1280, 1976.

59. J.E. Akin and W.H. Gray, Contouring on Isoparametric Surfaces Short Communications, Int, J. for Num. Meth. in Engng., Vol. 11, p. 1893-1897, 1977

60. W.H. Gray and J.E. Akin, An Improved Method for Contouring on Isoparametric Surfaces. Int. J. for Num. Meth. in Engng., Vol. 14. p 451-472 (1979).

61. J.L. Meek and G. Beer, Contour Plotting of Data Using Isoparametric Element Representation Short Communication, Int. J. for Num. Meth. in Engng., Vol. 10, No. 4, p. 954-957, 1976.

62. J.E. Akin and W.C.T. Stoddart, Plane Intersections and Contours for General Isoparametic Solids, Comp. and Structures, Vol. 10, p. 155-157, 1979. 1979.

63. A.E. Frey, C.A. Hall and T.A. Porsching, An Application of Computer Graphics to Three-Dimensional Finite Element Analysis, Comp. and Structures, Vol. 10, p. 149-154, 1979.

64. M.B. Stephenson and H.N. Christiansen, A Polyhedron Clipping and Capping Algorithm and Display System for three-Dimensional Finite Element Models, Computer Graphics, Vol. 9, No. 3, Fall 1975.

65. H.N. Christiansen and M.B. Stephenson, MOVIE.BYU--A Computer Graphics Software System, J. of the Technical Councils of ASCE, April 1979.

66. S. Murai and R. Tateishi, A Study on Image Processing for Results obtained for Finite Element Analysis, U.S.-Japan Seminar on Interdisciplinary Finite Set Analysis, Cornell U., Aug. 78.

67. W.E. Carroll, An Optimum Idealization in Discrete Element Analysis, Ph.D. Thesis, Virginia Polytechnic Institute and State Universiy, Blacksberg, VA, 1971.

68. G.M. McNeice and P.V. Marcal, Optimization of Finite Element Grids based on Mininum Potential Energy, Tech. Rep. No. 7, Brown University, Providence, RI, 1971.

69. D.J. Turcke and G.M. McNeice, A Variational Approach to Grid Optimization, Papers presented at the Int. Conf. on Var. Meth. in Mechanics, Southampton, 1971.

70. W.E. Caroll, Ramifications of Optimum Idealization of Geometry in Discrete Element Analysis, Proc. World Cong. of Finite Elements, in Structural Mechanics, Bournemouth, 1975.

71. I.C. Taig, Modeling and Interpretation of results in Finite Element Structural Analysis, Proc. World Cong. of Finite Elements in Structural Mechanis, Bournemouth, 1975.

72. D.J. Turke and G.M. McNeice, Guideline for Selecting Finite Element Grids Based on an Optimization Study, Comp. and Structures, Vol. 4, p 499-519, 1974.

73. W.E. Carroll, Inclusive Criteria for Optimum Grid Generation in the Discrete Analysis Technique, Comp. and Structues, Vol 6, p 333-337, 1976.

74. D.J. Turke and G.M. McNeice, Guidelines for Selecting Finince Element Grids based on an Optimization Study , Comp. and Structures, Vol. 4, p. 499-520, 1974.

75. R.J. Melosh and D.E. Killian, Finite Element Analysis to Attain a Prespecified Accuracy, Proc. 2nd Nat. Cong. on Computing in Structures, 1976.

76. Denayer, A., Automatic Generation of Finite Element Meshes Comp. and Structures, Vol. 9, p. 359-364, Oct. 78.

77. G.F. Carey, A Mesh-Refinement Scheme fo Finite Element Computations, Computer Methods in Applied Mechanics and Engineering, Vol. 7, p 93-105, 1976.

78. R.D. Bousquet and D.N. Yates, A Low Cost Interactive Graphics System for Large Scale Finite Element Analysis, Comp. and Structures, Vol. 3, p. 1321-1333, 1975.

79. V.A. Tischler and L.J.D. Bernier, Considerations for Develóping a General Finite Element Pre- and Postprocessing System, Structural Mechanics Computer Programs, W. Pilkey, K. Saczalski, H. Schaeffer (Eds.), U.P. of Virginia, 1974.

80. E.D. Herness and J.L. Tocher, Design of Pre- and Postprocessors, Str. Mech. Prog., W. Pilkey, K. Saczalski, H. Schaeffer (Eds.) U.P. of Virginia, 1974.

81. H.A. Kamel and M.W. McCabe, Geometry and Function Definition for Discrete Analysis and Its Relationship to the Design Data Base, Proc., Finite Int. Symp. on Comp-Aided Hull Surface Definition, (SCAHD 77), Annapolis, Md., Sept. 77.

82. H.A. Kamel, Design and Implementation of Interactive Engineering Software, U.S.-Japan Seminar on Interdisciplinary Finite Element Analysis, Cornell U., Aug 1978.

83. J.D. Foley, A Standard Computer Graphics Subroutine Package, Comp. and Structures, Vol. 10, p. 141-147, 1979.

84. ANSYS NEWS, Suanson Analysis Systems Inc., Fourth Quarter, 1978.

85. H.A. Kamel and M.B. Stephenson, Pre- and Postprocessing with GIFTS-5 and MOVIE-ARIZONA, Paper given at the United States Army Second Computer Graphics Workshop, Virginia Beach, Sept. 24-26, 1979.

DATE
FILMED

2-8